



# N7 SPACE

Evaluation of Rust usage in space applications by developing BSP and RTOS targeting SAMV71

Final Review Meeting 2024-01-30

# Agenda

- Project objectives summary
- Project achievements summary
- Demo
- Action statuses
- Risk Registry
- Schedule
- Deliverables
- Discussion

# Project objectives summary

“The proposed activity is to evaluate the usage of Rust programming language in space applications, by providing an RTOS targeting ARM Cortex-M7 SAMV71 microcontroller, a BSP (Board Support Package) and a Demonstration Application.”

## Tasks:

- Aerugo RTOS development
- SAMV71 BSP development
- Demonstration application development
- Creation of Rust Viability Report

# Project achievements summary

- RTOS implementation is completed
  - All requirements are covered
- BSP implementation is completed
  - All requirements are covered
- DEMO application implementation is completed
  - All required functionalities are exercised in the mission scenario
- Rust Viability Report is completed
  - Contains Aerugo and Rust review, conclusions, way forward, etc.

# DEMO

---

2024-01-09

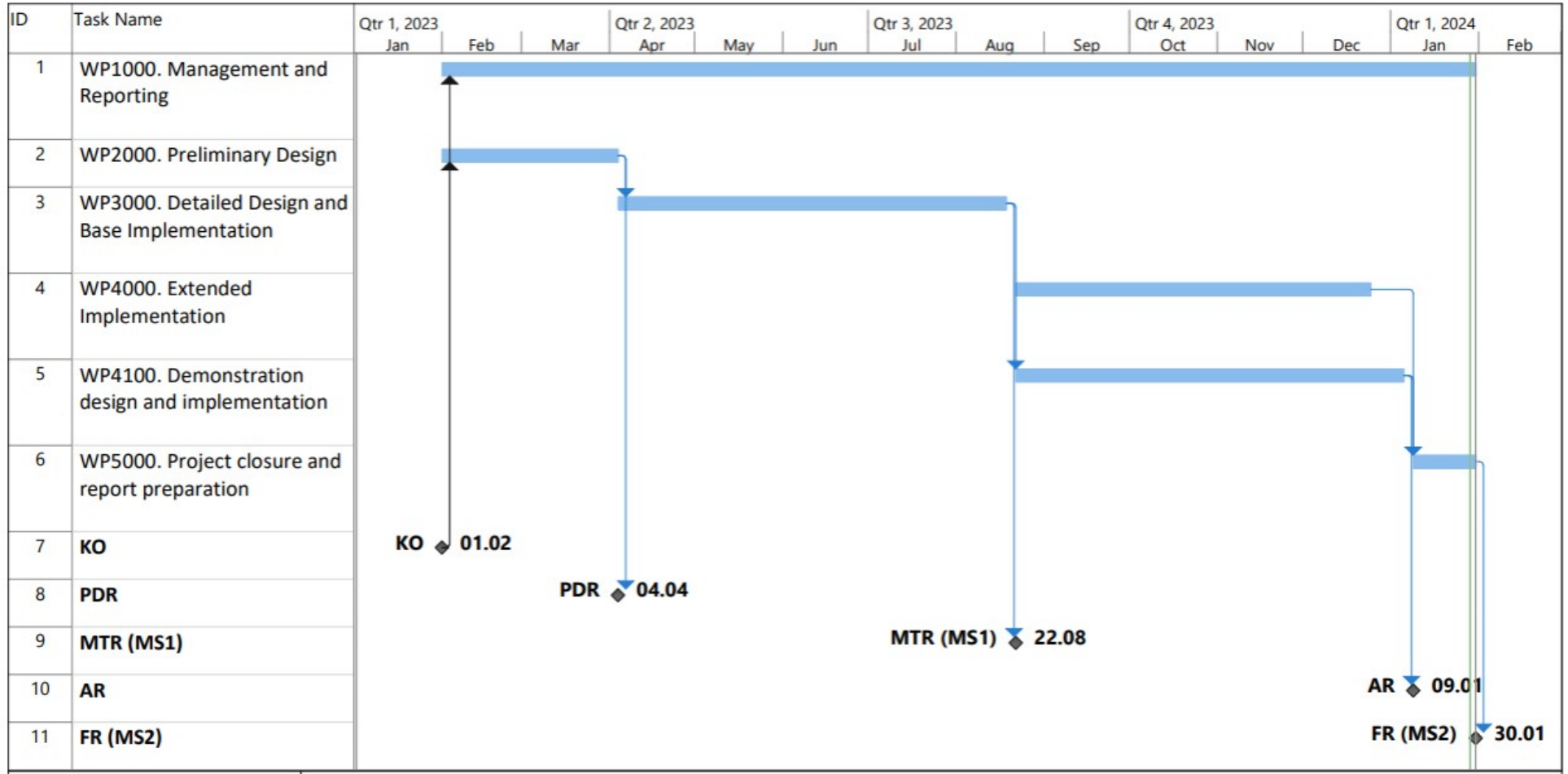
# Action statuses

Id	Origin	JIRA AI ID	Description	Actor(s)	Comment / Close-out Reference	Status
1	KO	<a href="#">RUSTOS-32</a>	Lessons-learned report to be included in RVR (long-term action).	N7S	Report was delivered in the FR data pack.	Closed
2	KO	<a href="#">RUSTOS-33</a>	Specify which data acquisition peripherals will be implemented.	N7S	List of peripherals that are to be implemented was created.	Closed
3	KO	<a href="#">RUSTOS-34</a>	Confirm the usage of RustDoc for SDD attachments generation.	ESA	Usage of RustDoc was confirmed.	Closed
4	KO	<a href="#">RUSTOS-35</a>	Document delivery method to be confirmed on ESA side.	ESA	Alfresco was chosen as the document delivery method.	Closed
5	PM1	<a href="#">RUSTOS-45</a>	Provide logins to Alfresco	ESA		Closed

# Risk Registry

Id	Title	Description	Risk Domain	Severity	Likelihood	Index	Impacted WPs	Action Summary and Status	Status
1	Compiler maturity	Target platform, SAMV71, chosen for this activity is, per Rust documentation, placed into a Tier 2. This can be thought as "guaranteed to build", which is assured by the official automated builds. It also means that the full standard library is supported on this target. Nevertheless, it's not guaranteed that all compiler tests passes, so it's not impossible to encounter some problems with building for this target.	Technical Cost Schedule	2	1	2	WP2000, WP3000, WP4000, WP4100	No problems with the toolchain for SAMV71 target was encountered during the implementation.	Closed
2	Architecture patterns enforced by Rust	Rust takes a unique approach to architecture patterns and default constructs comparing to other popular languages in this field. Although team members have experience with Rust, this still can impact coding patterns and increase development time.	Technical Cost Schedule	1	2	2	WP2000, WP3000, WP4000, WP4100	<p>Executor proof-of-concept created in WP2000 was used as a base for the design of the system architecture which was done before actual implementation mitigating potential risks in the future.</p> <p>BSP was designed in line with community standards. This helped to avoid potential traps and problems.</p> <p>No blocking issues were encountered during the implementation. All minor obstacles were easily resolved by taking different approaches, which didn't cost much time.</p>	Closed
3	Potential hardware issues	Driver development for embedded systems may encounter issues due to the complexity of testing (sometimes dependent on external devices), bottlenecks due to hardware availability or speed, unreliable hardware and connections, undocumented or under documented hardware bugs and unforeseen problems in re-used code. Although some existing libraries are available for re-use, they are not working completely out of the box and the scope of their functionalities aren't fully known. Depending on the usability and feature-set maturity of the libraries, the time needed to develop BSP could change.	Technical Cost Schedule	2	1	2	WP4000, WP4100	Aside for small bugs which were easily eliminated no blocking or serious problems were encountered during the implementation.	Closed

# Schedule





# Deliverables

- [SW1] Source code of the Rust SAMV71 BSP
  - Delivered via Git repository ([GitHub](#)) and uploaded the current version to Alfresco
- [SW2] Source code of the Rust operating system
  - Delivered via Git repository ([GitHub](#)) and uploaded the current version to Alfresco
- [SRS] Software Requirement Specification
  - Unchanged from the AR version
- [SUM] Software User Manual
  - Unchanged from the AR version
- [SDD] Software Design Document
  - Unchanged from the AR version

# Deliverables

- [SVSR] Software Validation Specification and Report
  - Delivered via Alfresco.
  - Contains:
    - Documentation of the delivered tests (RD)
    - Test reports (RDs)
    - Requirement coverage matrices (RDs)
    - Design and Inspection requirement coverage

## Coverage:

- Total requirement coverage is  $145/145 == 100\%$
- Number of requirements to be validated by test is 110
- Testable requirement coverage is  $110/110 == 100\%$

# Deliverables

- [ESR] Executive Summary Report
  - Delivered via Alfresco
- [PH] Photographic documentation
  - Delivered via Alfresco
  - Photos of the board used for developing and demonstration
- [FRP] Video presentation
  - Delivered via Alfresco
  - Video recording of the demonstration application following the mission scenario
- [RVR] Rust Viability Report / Final Report
  - Delivered via Alfresco
  - As discussed, this is considered a final report
  - Contains thoughts, conclusions, way forward, etc., summarizing output of this activity

# Discussion

- Any questions?

# THANK YOU FOR YOUR ATTENTION



Filip Demski  
[fdemski@n7space.com](mailto:fdemski@n7space.com)

Wojciech Olech  
[wojciech.olech@n7space.com](mailto:wojciech.olech@n7space.com)

Michał Kurowski  
[mkurowski@n7space.com](mailto:mkurowski@n7space.com)